International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2009)
Saratoga Springs, New York, May 3-7, 2009, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2009)

Rejection Sampling from a Spherical Harmonics Angular Flux Functional Expansion for a Discrete Ordinates to Monte Carlo Splice

S.C. Wilson and D.P. Griesheimer
Bettis Atomic Power Laboratory
P.O. Box 79
West Mifflin, PA 15122-5357
wilsonsc@bettis.gov; grieshei@bettis.gov

ABSTRACT

Splice methods offer the ability to accurately solve larger and more complicated problems than traditional solution techniques. In advanced splice methods, two (or more) distinct numerical methods are applied. Such methods allow users to increase local solution accuracy and reduce computational expense by judiciously selecting the method best suited to a particular region within the problem space. The primary challenge with developing effective splice methods of this type is the development of algorithms for coupling different solution techniques at the splice interfaces. This paper develops a method of representing angular flux data from a primary discrete ordinates solution as a function using spherical harmonics. Rejection sampling from the function is then used as a source in a secondary Monte Carlo solution. The three dimensional discrete ordinates radiation transport code PARTISN and the Monte Carlo radiation transport code MCNP were used to splice across a simple neutron transport problem exhibiting a forward-peaked angular flux distribution at the selected splice location. Functionality was added to MCNP to permit rejection sampling from spherical harmonics. Standalone global solutions from both codes are presented and compared to the spliced solution.

Key Words: rejection, harmonics, discrete ordinates, Monte Carlo

1. INTRODUCTION

1.1. Discrete Ordinates Splice Costs and Benefits

Splice techniques for discrete ordinates radiation transport problems typically require storage of angular fluxes along a user defined splice plane (or group of planes) at the discrete angles of the selected angular quadrature set. These angular fluxes may be used as a surface source in a secondary discrete ordinates problem. This type of splice technique can offer several advantages over a single global discrete ordinate problem. For example, large problems exceeding the capabilities of a particular computing system may be solved by "stringing" spliced solutions together. Additionally, interpolation options have been developed that permit users to specify different spatial mesh and angular quadratures for these secondary problems [1-3]. Interpolation options improve versatility by allowing users to coarsen or refine the spatial mesh, change the angular quadrature, and modify the spatial extent of the problem space (i.e., increased solution fidelity in regions of interest). However, discrete ordinates splice techniques require some overlap between the primary and secondary problem spaces, resulting in increased total computational and storage cost. Also, poorly specified splice problems may fail to accurately characterize important radiation paths and produce spurious scalar flux results.

Spliced discrete ordinates methods provide a variety of advantages, particularly with the interpolation options mentioned above, but require intelligent selection of splice planes and may increase computational cost. However, these requirements are often acceptable for otherwise-unsolvable problems. Simply splicing two or more discrete ordinates problems will not alleviate any weaknesses inherent in the discrete ordinates method. However, it is possible to ameliorate some weaknesses by appropriate use of splice interpolation techniques.

1.2. Relevant Aspects of Discrete Ordinates and Monte Carlo Methods

Discrete ordinates methods generally require less computational effort than Monte Carlo methods in regions of high total optical thickness. While discrete ordinates methods have their own unique problems (addressed below), they provide a deterministic scalar flux solution everywhere in the problem space. This advantage often makes the discrete ordinates method the solution technique of choice for large-scale shielding design applications.

Discrete ordinates solutions of the linearized Boltzmann transport equation may produce nonphysical ray effects in the scalar flux, particularly in regions comprised of weakly scattering media. This is typically considered the greatest weakness of the method. Other unfavorable aspects of the method include the possibility of negative flux solutions and discretization errors. Appropriate application of the discrete ordinates method, including adaptive solver methodologies, adequate meshing techniques and advanced angular quadrature sets, can substantially reduce the effect of these issues.

In particular, large voids and small streaming gaps tend to produce ray effects in discrete ordinate solutions. Increasing the order and sophistication of the angular quadrature set will ameliorate ray effects to a degree [2], but will not eliminate them entirely. In regions where severe ray effects are present, the scalar flux solution may become useless for design applications.

Because Monte Carlo solution techniques do not require discretization of the angular variable, Monte Carlo solutions do not exhibit the ray effects observed in discrete ordinates solutions. Unfortunately, Monte Carlo methods are typically more computationally expensive than discrete ordinates methods. However, an exception exists for regions comprised of few cells consisting of weakly interacting materials, where particles can stream for long distances between interactions. Therefore Monte Carlo methods can be used to efficiently analyze the types of large voids and streaming gaps that pose difficulties in discrete ordinates solutions.

1.3. Combining Discrete Ordinates and Monte Carlo Solution Strengths

This situation immediately suggests combining the two numerical methods to offset their respective weaknesses. Two options for combining the two methods exist: developing a standalone integrated hybrid code, or developing a splice method to convert discrete ordinates boundary source data to probability distributions appropriate for Monte Carlo sampling.

1.3.1. Hybrid Discrete Ordinates/Monte Carlo Code

Development of a new hybrid code with the option to solve user-specified spatial regions with a multi-group discrete ordinates method and other user-specified spatial regions with a fully continuous Monte Carlo method is attractive and daunting. Assuming these user-specified regions are defined with the combinatorial geometry featured in some Monte Carlo codes, one must include methods for discretizing arbitrary combinatorial geometry in a fashion suitable for the discrete ordinates solver included in the code. The discrete ordinates solver should be sophisticated enough to solve on an unstructured mesh, or risk losing fidelity to the true problem geometry. Parallelization of discrete ordinates solvers suitable for this degree of geometric sophistication is nontrivial, as optimized discrete ordinates parallel sweeping schemes often rely on brick rectilinear spatial meshes.

However, the advantages of a fully integrated hybrid code are numerous. Given optimal specification of discrete ordinates and Monte Carlo regions, a hybrid code could potentially solve a "real" complete problem faster and more accurately than either method could alone. Discrete ordinates ray effects would be eliminated by solving streaming regions with Monte Carlo. Monte Carlo variance reduction requirements would be substantially relaxed, as the discrete ordinates solver would handle optically thick regions of the problem space. The weaknesses of both methods could, in general, be avoided by simply calling the appropriate solver.

This discussion very briefly addresses just a few of the advantages and challenges of a hybrid radiation transport code. However, it also serves to make the important point that, while conceptually very attractive, developing a *full-featured* hybrid radiation transport code capable of solving the types and scope of problems MCNP or PARTISN solve would require a significant code development effort with a large commitment of resources, including experienced radiation transport method developers.

1.3.2. Discrete Ordinates/Monte Carlo Splice Code

Developing methods to splice from a discrete ordinates solution space into a Monte Carlo solution, and vice versa, is less elegant than an integrated approach and introduces all the disadvantages listed previously for splice problems. In general this means that the problem spaces for the two methods must overlap to some extent and, if the two (or more) problem spaces are strongly coupled (i.e., substantial transport occurs across splice boundaries in both directions), the solution must iterate among the separate spaces until a convergence criteria is satisfied. Furthermore, the configuration of the problem spaces is inherently restricted to whatever the constituent transport solvers support. These disadvantages notwithstanding, one avoids the large commitment of resources associated with developing a new radiation transport code. Additionally, the development and maintenance of the transport solution software is left to external development teams, leaving only the splice software to be maintained and updated.

1.4. Summary of Work

This paper develops the latter option for splicing between the discrete ordinates transport code PARTISN [4] and the Monte Carlo radiation transport code MCNP [5]. The splice option is developed only for fixed-source, time independent transport problems, and only from the discrete ordinates space into the Monte Carlo space. The reverse of this process (Monte Carlo to discrete ordinates) and integration of both options into an automated executive splice code is not addressed in this paper.

2. THEORY

2.1. Overview

The original work in this publication pertains largely to expansion of discrete ordinate angular fluxes in spherical harmonics and rejection sampling from the corresponding moments in Monte Carlo to closely reproduce the angular dependence of the incoming flux. It is also necessary to incorporate methods of sampling from discrete cumulative probability distributions for other variables (i.e., space and energy). We briefly review the theoretical basis of these methods below, and examine methods of producing spherical harmonics moments from discrete data sets and rejection sampling from continuous functions.

2.2 Sampling

Two methods of sampling from probability distributions are employed in the PARTISN-to-MCNP splice code. Standard sampling from a discrete probability distribution is employed in the energy and spatial dimensions. Rejection sampling from evaluated spherical harmonics is employed in the angular dimension.

2.2.1. Sampling from discrete cumulative distribution functions

For one-dimensional probability distributions, we have the following:

$$p_i = p(x_i) \tag{1}$$

defined at the discrete bin midpoints x_i. The discrete cumulative probability distribution defined at the bin midpoints may be approximated by:

$$P_i = \left[\sum_{n=1}^{i-1} P_n \right] + p_i \quad for\ i = 1, 2, \ldots N \tag{2}$$

where N is the number of discrete data points in the probability distribution p. If p is not appropriately normalized to restrict P to (0,1), P must be normalized with normalization constant P_{N-1} prior to sampling. Given the normalized discrete cumulative probability distribution P, the simplest way of sampling requires the generation of a uniformly distributed random number:

$$\xi \in (0,1) \tag{3}$$

We then search *P* to find the first discrete bin *i* that satisfies $P_i \geq \xi$. We may select the bin average $(p_i + p_{i+1})/2$ as the randomly generated value of *p*, sample uniformly across the selected bin, or choose a more advanced method of sampling across the discrete bin [7].

Sampling in energy in the PARTISN-to-MCNP splice code under discussion is a straightforward application of the process described above. Sampling in the two spatial dimensions requires transforming a two-dimensional array of discrete probabilities into a one-dimensional list. As long as one retains the details of the mapping from multiple dimensions to a single dimension, the above method may be applied.

2.2.2. Rejection sampling from continuous probability distribution functions

If a continuous probability distribution is available and construction of the cumulative probability distribution is prohibitively expensive or otherwise undesirable, one may rejection sample directly from the probability distribution function. Given a probability distribution $p(x)$ on domain (a, b) with a range $(0, p_{\max})$, we generate two independent, uniformly distributed random numbers:

$$\xi_1 , \xi_2 \in (0,1) . \tag{4}$$

Then we check the following condition:

$$p\left(\xi_1 \cdot (b-a) + a\right) \leq (\xi_2 \cdot p_{\max}) \tag{5}$$

If the above condition is true, we accept the value of ξ_1 as a sample from $p(x)$. If the condition is false, we reject the value of $p(\xi_1)$ and generate another two random numbers. The process is continued until acceptance.

Extension of this method to multiple dimensions is straightforward. For a three-dimensional probability distribution function $p(x,y,z)$, we use exactly the process above, except with four random numbers instead of two. Expansion to *n* dimensions requires $(n + 1)$ random numbers: *n* random numbers to evaluate the function at a random point and one random number to compare to the value of the function.

This technique becomes less efficient the more "peaked" the distribution function. In one dimension the efficiency of the technique is the ratio of the area under the normalized probability distribution $p(x)$ to the area of the unit box:

$$r = \frac{\int_a^b p(x)\,dx}{(b-a)\,p_{\max}} \tag{6}$$

If this number is very small ($r << 1$), most pairs of random numbers (ξ_1, ξ_2) will result in rejection. Alternatively, if Eq. (6) evaluates to a number close to unity, the method is highly efficient and almost all pairs (ξ_1, ξ_2) will result in acceptance.

Rejection sampling is used in the PARTISN-to-MCNP splice code to reproduce the angular dependence of the PARTISN boundary source. The continuous function in three angular variables (μ,η,ξ) is evaluated from constructed spherical harmonics moments.

2.3 Spherical Harmonics

The angular flux acquired from a discrete ordinates solution is evaluated only at the discrete angles of the applied angular quadrature. In generating a source for a Monte Carlo calculation, this data could be integrated and binned on the unit sphere in some fashion to permit discrete sampling. Alternatively, the data may be expanded in a series of orthogonal functions, providing a continuous function suitable for rejection sampling. The latter is the approach adopted in this paper.

2.3.1. Spherical Harmonics Functions

The spherical harmonics functions are defined in terms of associated Legendre functions, which are in turn defined in terms of Legendre polynomials. The Legendre polynomials are defined by Rodrigues' formula [6]

$$P_\ell(x) = \frac{1}{2^\ell \ell!}\frac{d^\ell}{dx^\ell}(x^2-1)^\ell \tag{7}$$

The associated Legendre functions are defined in terms of P_n [6]:

$$\begin{aligned} P_\ell^m(\mu) &= (-1)^m(1-\mu^2)^{m/2}\frac{d^m}{d\mu^m}P_\ell(\mu)\,, \\ P_\ell^{-m}(\mu) &= (-1)^{|m|}\frac{(\ell-|m|)!}{(\ell+|m|)!}P_\ell^{|m|}(\mu)\,, \\ |m| &\le \ell \end{aligned} \tag{8}$$

Generating functions and recursive relations provide simple, efficient ways of numerically evaluating the Legendre polynomials and associated Legendre functions of arbitrary order at μ on (–1, 1) [6,7].

The spherical harmonics functions are orthogonal functions on the unit sphere. They are defined as:

$$Y_{\ell,m}(\mu,\theta)=\left[\frac{(2-\delta(m,0)(\ell-m)!}{(\ell+m)!}\right]^{1/2} P_\ell^m(\mu)\cos(m,\theta)\,,$$
$$Y_{\ell,-m}(\mu,\theta)=\left[\frac{2(\ell-|m|)!}{(\ell+|m|)!}\right]^{1/2} P_\ell^{|m|}(\mu)\sin(|m|,\theta) \tag{9}$$

where $\delta(m,0)$ is the Kronecker delta, μ is the cosine of the polar angle and θ is the azimuthal angle.

Taking advantage of the spherical harmonics orthogonality on the unit sphere, we may construct moments, defined in terms of the spherical harmonics and some function $\psi(\mu,\theta)$ as:

$$\phi_{\ell,m} = \int_{2\pi} d\theta \int_{-1}^{1} d\mu\, \psi(\mu,\theta) Y_{\ell,m}(\mu,\theta) \tag{10}$$

which allow expression of the function $\psi(\mu,\theta)$ in an infinite series of spherical harmonics:

$$\psi(\mu,\theta) = \sum_{\ell=0}^{\infty} \frac{2\ell+1}{4\pi} \sum_{m=-\ell}^{\ell} \phi_{\ell,m}\; Y_{\ell,m}(\mu,\theta) \tag{11}$$

2.3.2. Least Squares Data Fitting to Spherical Harmonics

Given a discrete data set of the form:

$$\psi_i = \psi(\mu_i,\theta_i)\,,\; i=1,2,...,N \tag{12}$$

defined on the unit sphere, we may fit the data to a truncated spherical harmonics expansion and interpolate or extrapolate to points on the unit sphere outside the discrete locations (μ_i, θ_i) of the original data [3]. Since we have only *N* data points available, the order of the truncated spherical harmonics expansion, *L*, must satisfy the following condition:

$$(L+1)^2 \le N \tag{13}$$

The resulting spherical harmonics expansion has the form:

$$\psi(\mu_i,\theta_i) \approx \sum_{\ell=0}^{L} \frac{2\ell+1}{4\pi} \sum_{m=-\ell}^{\ell} \phi_{\ell,m}\; Y_{\ell,m}(\mu_i,\theta_i)\,,\; for\;\; i=1,\,2,\,\ldots,\,N \tag{14}$$

This expression may be represented as a linear system of equations:

$$\underline{\underline{A}}\,\underline{\phi} = \underline{\psi} \tag{15}$$

where, given $M = (L+1)^2$, A is an $N \times M$ matrix, ϕ is an $M \times 1$ column vector and ψ is an $N \times 1$ column vector. In this equation ψ is a vector of known data, ϕ is a vector of unknown spherical harmonics moments, and A has the form:

$$\begin{aligned} &\underline{\underline{A}}_{i,j} = c_{\ell_j} Y_{\ell_j, m_j}\left(\mu_i, \theta_i\right), \text{where} \\ &c_\ell = \frac{2\ell+1}{4\pi}, \\ &\ell_j \text{ is of the form } (0,1,1,1,2,2,2,2,2,\ldots), \text{ and} \\ &m_j \text{ is of the form } (0,-1,0,1,-2,-1,0,1,2,\ldots) \end{aligned} \tag{16}$$

We now solve for ϕ such that the L_2 norm of the model equation, $\left\| \underline{\psi} - \underline{\underline{A}}\underline{\phi} \right\|_2^2$, is minimized. This may be accomplished by any of a variety of least squares solution techniques.

Once the vector ϕ of spherical harmonics moments is known, one may evaluate the truncated spherical harmonics expansion at data points outside the original set by constructing a new A matrix using the desired points on the unit sphere and multiplying by the moments vector:

$$\underline{\psi}_{\text{new}} = \underline{\underline{A}}_{\text{new}} \underline{\varphi} \tag{17}$$

It is worth noting that the spherical harmonics functions are well-behaved, continuous functions on the unit sphere. Attempts to fit data which do not match this description to a truncated spherical harmonics expansion will typically exhibit partial or even total failure.

3. IMPLEMENTATION

3.1. Overview

In constructing a suitable MCNP5 source from a PARTISN boundary source, we begin by reading the PARTISN boundary source. The format of this file is conveniently available in the PARTISN documentation [4]. While parsing the PARTISN boundary source file, we perform the following steps concurrently:

1) Integrate the total incoming partial current over the entire boundary source in each energy group;
2) Integrate the energy-dependent partial current over the surface of each spatial cell;
3) Fit the energy and space-dependent angular fluxes to a set of spherical harmonics moments;
4) Store the partial current and spherical harmonics data to disk.

After the PARTISN angular flux data has been parsed and stored in binary format, the data is spatially decomposed for parallelization and written in ASCII to facilitate transfer across platforms. Once the data and problem specifications have been decomposed and stored in this fashion, a modified version of MCNP 5.140 can read the processor-dependent source files, store them in memory and sample from the data to begin particle histories. This process is explained in detail below.

3.1.1. PARTISN Boundary Sources

A PARTISN boundary source file contains a variety of problem-dependent information, including spatial mesh details, angular quadrature information and, most significantly, the angular flux data for each cell in every energy group.

It is worth clarifying exactly where (spatially) the PARTISN angular flux values are defined in the problem geometry. The user specifies the Cartesian mesh plane (splice plane) where the angular fluxes are desired. The mesh is defined only on cell edges, not cell centers. However, observing the boundary source from the normal perspective, the angular fluxes are located at the center of each spatial mesh cell. By way of example: if the user has requested a +z source at plane 40, the angular fluxes defined at spatial mesh 40 (i.e., at the positive-going cell *edge* of cell 39) in the z-dimension will be written to file, but will still be cell-centered in the x and y-dimensions.

In a typical three-dimensional *xyz* discrete ordinates calculation, the angular flux will be subdivided into eight octants, one for each combination of (±x, ±y, ±z). The PARTISN boundary source file contains only those four octants with non-zero partial currents in the source direction. If one requests a +z source, only the four (±x, ±y, +z) octants will be stored in the angular flux file.

3.1.2. Energy Probability Distribution and Sampling

We must provide probability information to the modified MCNP 5.140 program to reproduce the discrete ordinates angular flux behavior in the Monte Carlo source. The first quantity of interest is the energy group-dependent partial current integrated over the entire spatial extent of the boundary source. It must be sampled first because the spatial and angular behaviors are both energy-dependent. As we have fluxes available only at discrete locations, we perform the following summation to approximate the integral:

$$J(E_g) = \frac{1}{c} \sum_{i=1}^{D1} \sum_{j=1}^{D2} \sum_{r=1}^{N} \psi_{i,j,r} n_r A_{i,j} wgt_r \qquad (18)$$

where D_1 is the number of cells in the first spatial dimension, D_2 is the number of cells in the second spatial dimension, N is the total number of angles in the cell (i.e., four times the number of angles per octant), *wgt* is the angular quadrature weight and n is the dot product of the direction (μ_r, θ_r) of angular flux $\psi_{i,j,r}$ with the unit vector normal to the boundary source plane. We multiply by the cell area $A_{i,j}$ to approximate the differential $dx\,dy$.

The discrete values $J(E_g)$ provide energy probability values at the midpoint of each energy bin E_g. These discrete probabilities may be used to build and normalize cumulative probability distributions as described in Section 2.2.1. The constant c in Eq. 19 represents the normalization constant.

In sampling from the resulting cumulative probability distribution, it is important to note that, for most groups, energy values may be generated uniformly across the selected energy group bin. However, this constant spectrum assumption does not hold in cases where the thermal energy range is treated as a single energy group. In this case, a Maxwellian distribution is assumed within the thermal group and sampled appropriately. Otherwise uniform random sampling across the energy bin appears sufficient to convert the group-wise discrete ordinate results into a continuous energy Monte Carlo source term, provided there are enough energy groups to adequately capture the energy-dependent behavior of the angular flux.

3.1.3. Spatial Probability Distribution and Sampling

After the energy group has been selected and the continuous energy E generated, we calculate the within-group partial current for each spatial cell to construct a probability distribution:

$$J(x_i, y_j) = \frac{1}{c}\sum_{r=1}^{N} \psi_{i,j,r} n_r A_{i,j} wgt_r \tag{19}$$

where the group index has been dropped. We retain multiplication by cell area in order to obtain a total source emission rate (particles per second) from the cell. These calculated partial currents in two-dimensional matrix form are mapped to a one-dimensional vector of dimension ($D_1 \times D_2$). With a one-dimensional list of discrete probabilities available, the method to generate a normalized discrete cumulative probability distribution described in Section 2.2.1 applies directly. Constant c above represents the normalization constant. Once the normalized cumulative probability distribution is generated, we may sample from it in the same way we sampled for energy E.

Once a discrete spatial cell index has been selected, we invert the mapping to one dimension and acquire the two-dimensional integer coordinates of the cell, (j,k). The spatial mesh information (i.e., boundaries) for cell (j,k) is available from the parsed PARTISN boundary source file. Cell (j,k) is mapped to a (1×1) square, and continuous coordinates (x,y) on the unit square are generated by uniform sampling. The coordinates on the unit square are mapped back to the true cell dimensions, providing continuous coordinates (x,y) on the true cell geometry suitable for initiating a Monte Carlo source particle. The third spatial coordinate, z, is simply the location of the boundary source plane along the Cartesian axis. Uniform sampling within a selected cell (j,k) assumes the discrete ordinates mesh is fine enough to adequately approximate a continuous spatial treatment.

3.1.4. Angular Probability Distribution and Sampling

Given a selected energy bin and spatial cell, we may now use the energy and space-dependent spherical harmonics moments to correctly generate direction cosines via rejection sampling for the source particle flight direction, completing the source generation process. The pseudo-code for the algorithm is included in Table I.

The subroutine detailed in Table I accepts the spherical harmonics moments for a particular energy group and spatial cell, determined prior to calling the routine. The routine generates a random triplet of direction cosines (μ,η,ξ) on the correct half-sphere (recall the boundary source is defined only for forward-going partial currents) and evaluates the spherical harmonics moments at that point. The result is then normalized to the maximum angular flux from the discrete ordinates angular fluxes to map the range of the function to (0,1). The normalized result is compared to a randomly generated number on (0,1), and accepted or rejected based on the result of that comparison. If accepted, the routine returns the generated triplet of direction cosines to be used as the flight direction for the source particle. If rejected, the routine iterates the number of rejections and restarts the process.

The function maximum is set to the maximum detected angular flux within the generated energy group and spatial cell, recorded to disk during boundary source parsing. Since this discrete group of angular fluxes represents only a subset of the continuous angular flux, it is very likely the "true" angular flux maximum lies outside this subset, resulting in an incorrect normalization prior to testing for rejection. More angles in the discrete ordinates angular quadrature rule will reduce the likelihood of incorrect normalization introducing significant error to the Monte Carlo calculation.

In the rejection test conditional, note that a check is made to ensure the positivity of the generated function value. In the case of a particularly forward-peaked or otherwise poorly behaved (i.e., one where negative fluxes have been set to zero) angular flux distribution, the least squares fit of the angular flux data to a set of spherical harmonics moments will necessarily be imperfect. The fit will accurately reproduce the flux behavior in the solid angle region where the angular flux magnitude is relatively large and well-behaved, but may produce small, non-physical oscillations in the solid angle region where the angular flux is poorly characterized. These oscillations may produce negative fluxes in a fashion somewhat analogous to Legendre fits of forward-peaked scattering cross-sections. These negative fluxes would not pass the rejection test, but it is worth noting their existence explicitly.

Table I. Pseduo-code outlining the procedure for sampling from a probability density function defined as a series of spherical harmonic terms.

```
subroutine RejectionSample(NumMom,FuncMax,FluxMomLocal,MuOut,EtaOut,XiOut,Rejects)

integer,intent(in) :: NumMom
real,intent(in) :: FuncMax
real,intent(in),dimension(NumMom) :: FluxMomLocal

real,intent(out) :: MuOut,EtaOut,XiOut
integer, intent(out) :: Rejects

real :: R1,R2,R3,Ord,OrdNormed,mugen,phigen,thetagen,etagen,xigen

Rejects = 0

do

        generate random number R1
        generate random number R2
        generate random number R3

! acquire coordinates and direction cosines for a point on the unit sphere using random
! numbers 1 and 2

        mugen = R1*(2.0)-1.0
        phigen = R2*2.0*PI
        thetagen = ACOS(mugen)
        etagen = SIN(thetagen)*COS(phigen)
        xigen = SIN(thetagen)*SIN(phigen)

! take the absolute value of one direction cosine to restrict the  generated point to the
! appropriate half-sphere

        xigen = ABS(xigen)

! call a subroutine to evaluate the local spherical harmonics moments at the triplet generated
! above and store the result to a local variable (Ord)

        call EvaluateMoments(mugen,etagen,xigen,NumMom,FluxMomLocal,Ord)

! normalize the result to the maximum angular flux prior to testing for rejection

        OrdNormed = (Ord) / (FuncMax)

! test for rejection using random number R3

        if (R3 <= OrdNormed .AND. Ord >= 0.0) then

! Accept
                MuOut = mugen
                EtaOut = etagen
                XiOut = xigen
                exit
        else

! Reject
                Rejects = Rejects + 1
        end if

end do

end subroutine RejectionSample
```

The partial current was used to generate the energy and space probability distributions, but the angular flux is used to construct spherical harmonics moments for the rejection sampling routine. As stated previously, the spherical harmonics functions are well-behaved, continuous functions defined on the unit sphere. In situations where one encounters heavily forward-peaked angular fluxes, a distribution constructed from the dot product of the angular flux with the unit vector normal to the boundary source plane (i.e., the forward-going component of the angle-dependent source) may be substantially *more* forward-peaked. This can pose difficulties in accurately fitting the source data to a set of spherical harmonics moments. Therefore, the angular flux is used to generate the moments and the direction cosines. However, to correctly reproduce the source, one must assign a particle weight equal to the dot product of the flight direction and the unit vector normal to the boundary source:

$$wgt = (\mu \hat{i} + \eta \hat{j} + \xi \hat{k}) \cdot \hat{n} \tag{20}$$

where (i,j,k) are Cartesian unit vectors and $\hat{n}$ is the unit vector normal to the boundary source plane. In practice, due to a PARTISN restriction, the boundary source planes in this application are all perpendicular to a Cartesian axis. This means the weight will be one of the three generated direction cosines, depending on the orientation of the boundary source.

With (μ,η,ξ) correctly generated by the rejection routine, and (E,x,y,z) generated previously, the seven phase space variables required to generate a Monte Carlo source particle, along with the correct particle weight, have been generated.

3.2. MCNP5 Code Modifications

3.2.1. Outline of Required Modifications and Additions

While the main focus of this paper does not pertain to code development details, a broad list of required additions and modifications is provided for reference. Modifications to the code are minimized in order to ensure reduced effort in porting to new versions of MCNP. The only modifications are:

- a call to the MCNP MPI implementation in the “main” subroutine to acquire the number of processors and the rank of the local processor;
- a call in the MCNP “main” subroutine to the IO routine of the splice code to read in the ASCII source files containing partial currents and spherical harmonics moments
- a call to the sampling routines in the “source” subroutine included for user modification in the MCNP source code

Additions to the code are more extensive, but should not require modification when moving to new versions of MCNP. Note the MCNP compilation dependency list and makefile must be modified to include these additions:

- Splice global variables module
- Spherical harmonics module

- IO module
- Sampling module
- Driver routine

Exterior to the MCNP code is the PARTISN boundary source parser. This software calculates and writes to disk the partial currents and spherical harmonics moments as outlined previously, as well as other problem dependent information. All the code developed for this application adheres to the Fortran 90 standard.

3.2.2. Spatial Decomposition of the MCNP Source File

Boundary source files written by PARTISN tend to be quite large, depending on the problem specifications. While expansion of the angular fluxes in spherical harmonics moments substantially reduces the data storage requirement, it is still possible to overwhelm the storage capacity of a single master processor. Furthermore, funneling all the IO operations through the master processor creates an unnecessary bottleneck in the code execution. For this reason, the single-processor PARTISN boundary source parser will write a separate source file for each slave processor in the MCNP calculation. The parallel decomposition is carried out in the two spatial dimensions, giving each processor in the MCNP problem a small subset of cells from which to sample.

4. RESULTS

4.1. Summary

As the developed splice code splices in one direction only (i.e., discrete ordinates to Monte Carlo, not vice versa), it was necessary to test the code on a weakly coupled splice problem. For simplicity a two-region splice was selected. The problem source is located entirely in the primary problem, and particles transported from the secondary problem to the primary problem do not return to the secondary problem in numbers sufficient to noticeably alter scalar flux results.

4.1.1. Model Problem

The problem under analysis was selected to provide a forward-peaked angular flux distribution and demonstrate the ability of the PARTISN-to-MCNP splice code to accurately reproduce anisotropic flux distributions. The problem also satisfies the weak coupling criteria mentioned above.

As shown in Figures 1 and 2, the problem geometry consists of a fission source in water followed by a large air void, then an asymmetric steel grid in water. There is 10 cm of air following the steel grid. With the exception of the steel grid, the materials in the problem are uniform in the X-Y plane. The problem selected is known as the “tic-tac-toe” model for reasons evident in Figure 2. The entire problem was run in both PARTISN and MCNP, and then run with the splice code from a PARTISN primary region to an MCNP secondary region.

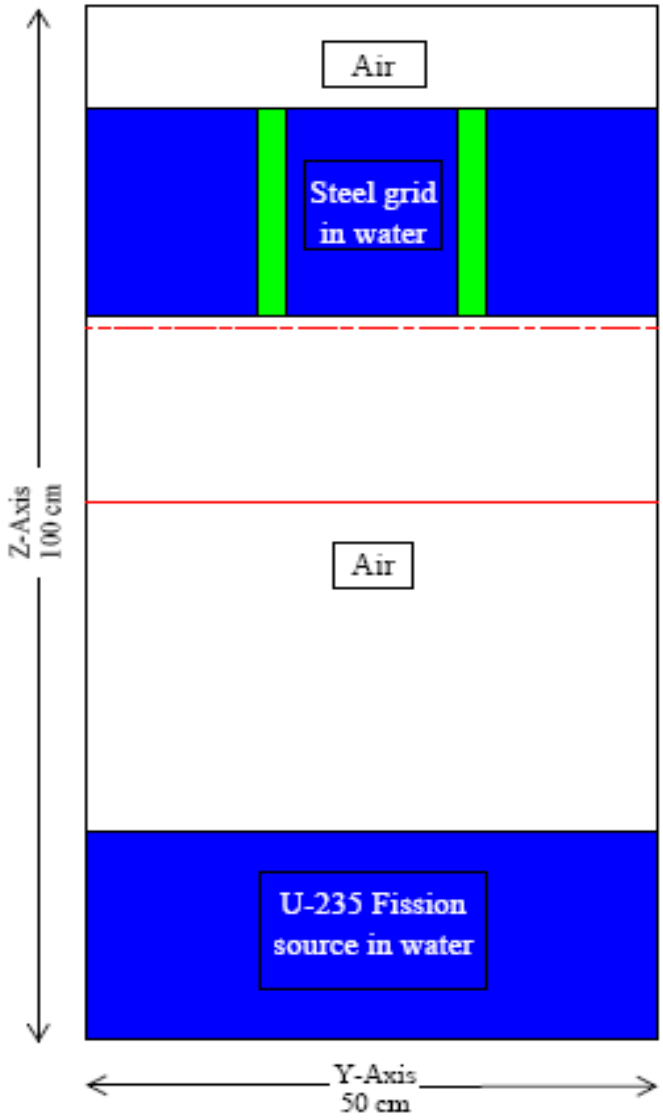

Figure 1. Top-down view of model problem geometry.

4.1.2. Discrete Ordinates Problem Summary

The PARTISN model of this problem space is able to exactly reproduce the model geometry, as there are no curvilinear bodies. The problem extent is 40 cm in the *x*-dimension, 50 cm in the *y*-dimension and 100 cm in the *z*-dimension. The spatial mesh is a uniform 0.5 cm everywhere in the problem space, resulting in mesh dimensions of 80 cells in the *x*-dimension, 100 cells in the *y*-dimension and 200 cells in the *z*-dimension. The total cell count is 1.6 million. Fifty-eight neutron energy groups are used to discretize the energy variable, with a single group comprising the thermal region. Gamma radiation is not treated in this problem. The problem was run with two QR quadrature sets [2]. The number of quadrature angles per octant ranged from 36 to 70.

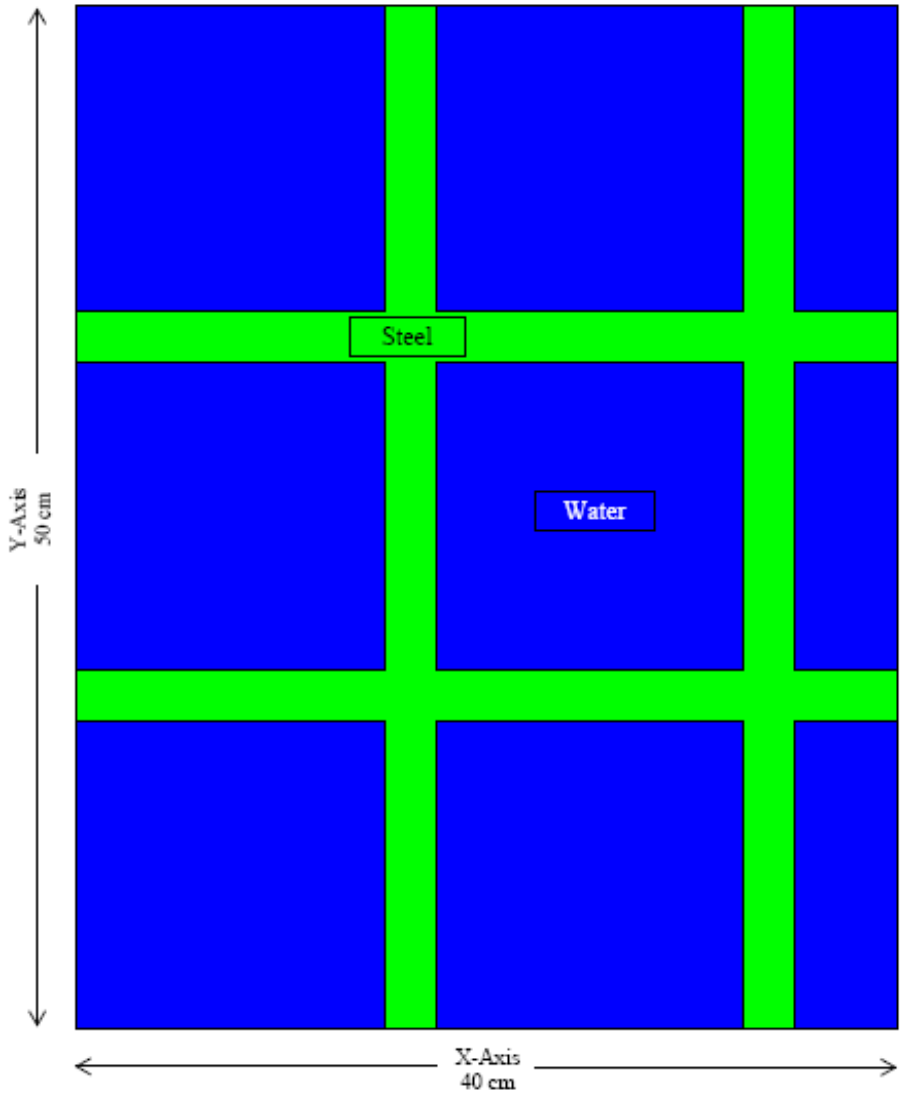

Figure 2. X-Y slice at Z = 75 cm.

The neutron source is defined in terms of the fifty-eight energy groups, and closely approximates a continuous U-235 fission spectrum. The source is emitted isotropically in angle and is uniform in space throughout the source extent. The boundary conditions for the problem are all vacuum. This is important in producing the forward-peaked angular distribution at the splice plane, as the angular fluxes in the directions of the *x* and *y* boundaries have largely leaked from the problem space prior to reaching the splice plane location.

4.1.2. Monte Carlo Problem Summary

The MCNP geometry of the problem consists entirely of boxes defined by combinations of simple planes, and reproduces the true geometry exactly. The source is defined in the same way the PARTISN source is defined: fifty-eight energy bins approximating a U-235 fission spectrum,

emitted isotropically in angle and uniformly in space throughout the source extent. The S(α,β) thermal neutron treatment for light water at room temperature is turned on. Cell-based weight windows are employed to improve computational efficiency. All boundary conditions are vacuum.

The MCNP tally structure is set up to acquire plane-integrated scalar fluxes every 0.5 cm, which is suitable for comparison to plane-averaged scalar fluxes from the PARTISN problem. A cosine tally is set up at the splice plane location to facilitate comparison of the angular distributions between the global MCNP solution and the splice MCNP solution.

4.1.3. Splice Problem Summary

A front-going (+z) splice boundary source is requested at z = 50 cm from a truncated PARTISN problem solving from z = 0 cm to z = 70 cm. In Figure 1, the splice plane is taken at the solid red line in air, and the primary PARTISN problem is solved from z = 0 cm to the dashed red line. It is necessary to solve past the splice plane (i.e., overlap the problem spaces) to avoid corrupting the front-going partial current with the artificial vacuum boundary in the truncated problem.

The boundary source file is parsed by the PARTISN-to-MCNP splice code. The splice code writes a user-defined number of spatially decomposed ASCII-formatted source files suitable for use by an MCNP executable modified per Section 3.

The secondary MCNP problem is run using these source files from z = 50 cm to the end of the spatial extent of the problem at z = 100 cm. This is the only source defined in the MCNP problem. The tally structure for the MCNP problem is identical to that of the full MCNP solution, except it is defined only for z = 50 cm to z = 100 cm.

A normalization factor is applied to the MCNP results to allow meaningful comparison to the discrete ordinates results. For the global results, the normalization constant is the volume integrated source from the discrete ordinates calculation. For the spliced results, the normalization constant is the ratio of the total partial current at the splice plane calculated during boundary source parsing to the partial current tallied at the splice plane in the secondary MCNP calculation.

4.2. Selected Results

Comparison of results summarizing the energy, spatial and, in particular, angular behavior with the three solution methods provides an effective measure of the accuracy of the developed splice method.

4.2.1. Spatial Scalar Flux Comparison

The differences between plane-averaged scalar flux solutions as a function of z-position are illustrated in Figure 3. The blue line (triangular markers) shows the percentage difference

between the global PARTISN and global MCNP solutions. At the z = 100 cm boundary the splice solution is within 1% of the global MCNP solution.

Figure 4 shows the plane-averaged scalar flux through the *z*-dimension of the splice region. It is interesting to note in both Figure 3 and Figure 4 the spliced solution tracks the global Monte Carlo solution more closely than it does the discrete ordinates solution, despite the discrete nature of the source in the splice problem. This indicates that continuous sampling of sufficiently finely discretized data will generate results in close agreement with a fully continuous solution.

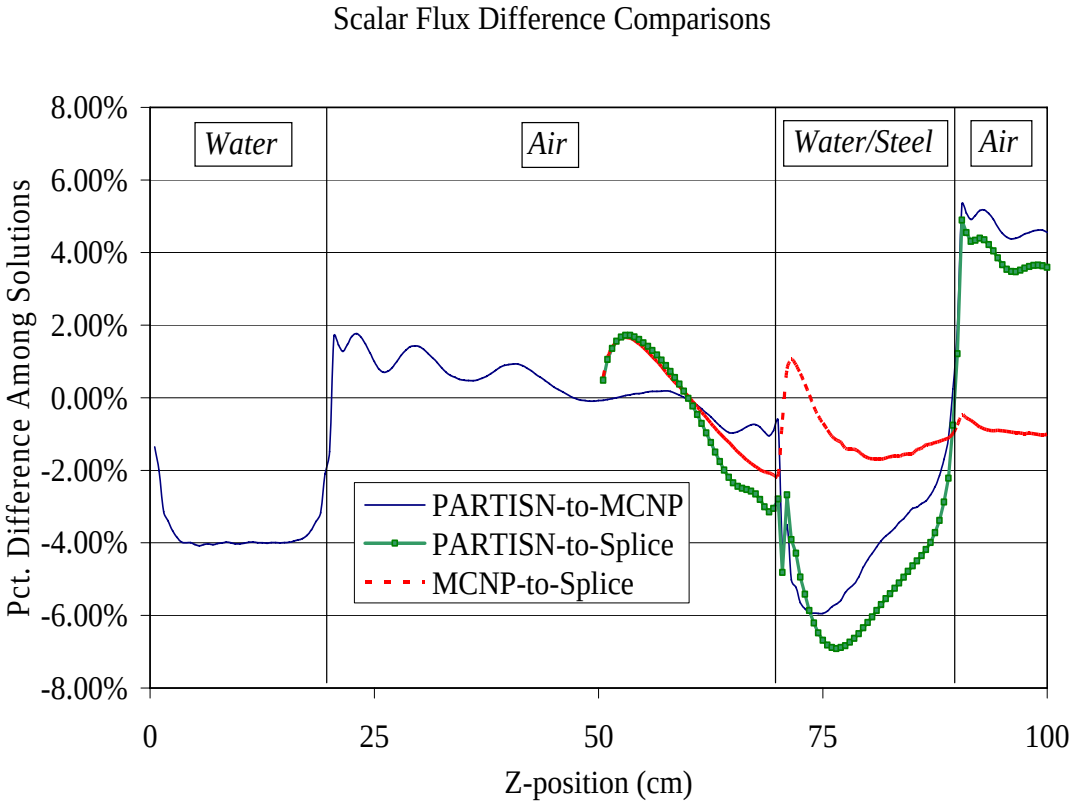

Figure 3. Scalar flux solution differences.

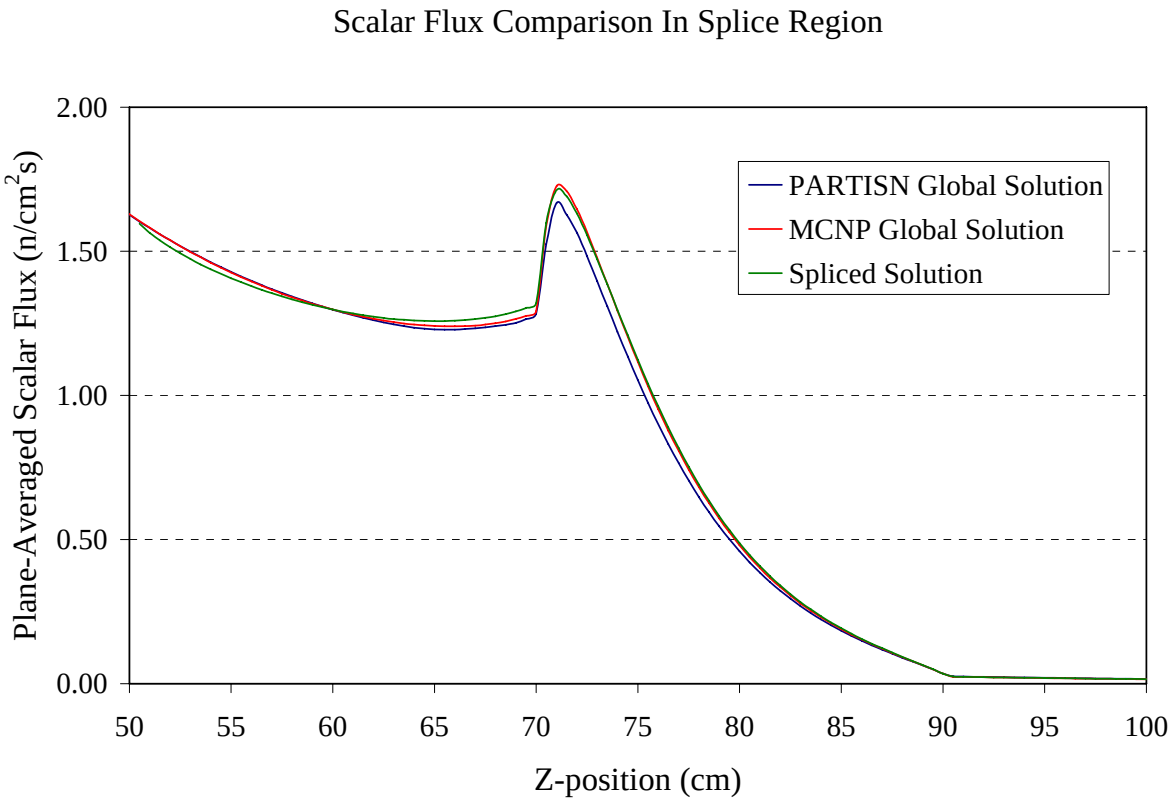

Figure 4. Scalar flux solutions in the splice region.

4.2.2. Energy Distribution Comparison

Capturing the energy distribution of the flux at the splice plane is essential when moving between numerical methods utilizing different energy representations. Figure 5 shows two energy-binned MCNP flux tallies taken at the splice plane: one from the spliced solution, the other from the global MCNP solution. The two distributions are virtually indistinguishable.

The same holds for the distribution past the steel grid region, shown in Figure 6, except for the thermal group. The difference between the spliced and global Monte Carlo solutions in the thermal energy bin after transport through 20 cm of steel and water is approximately 2%. It will require further study to isolate the cause of this small difference, but this result is certainly adequate for many applications.

We did not explicitly compare to the energy distribution of the PARTISN global solution, as it matched the MCNP global solution (binned into identical energy bins) at the splice plane.

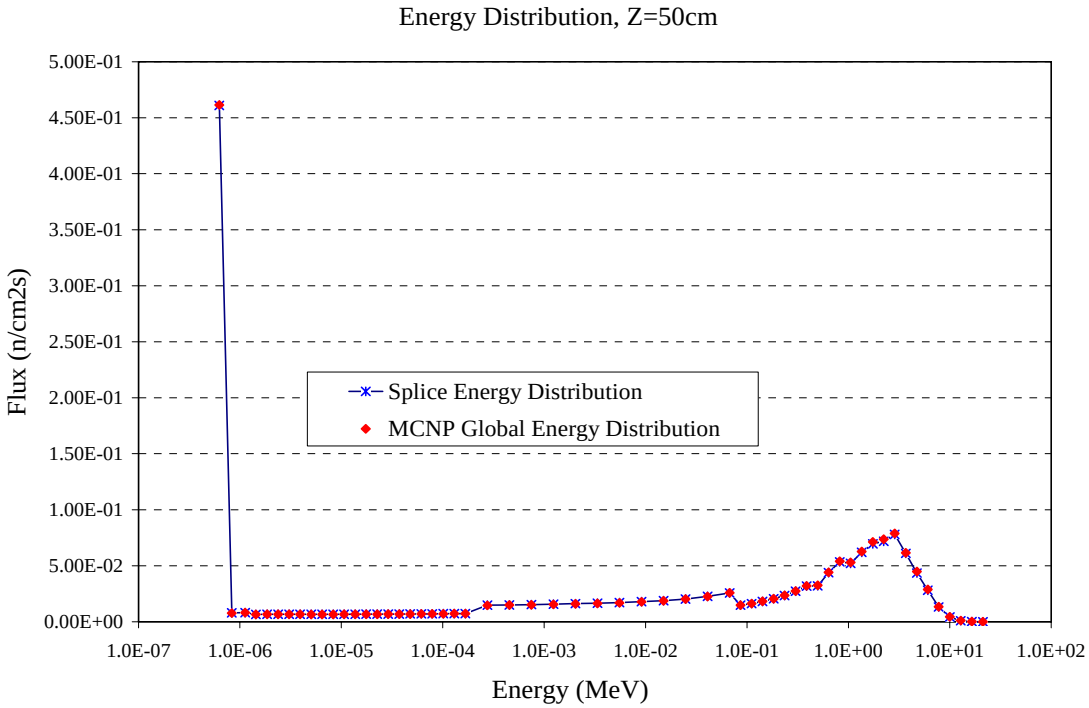

Figure 5. Energy distribution comparison at the splice plane.

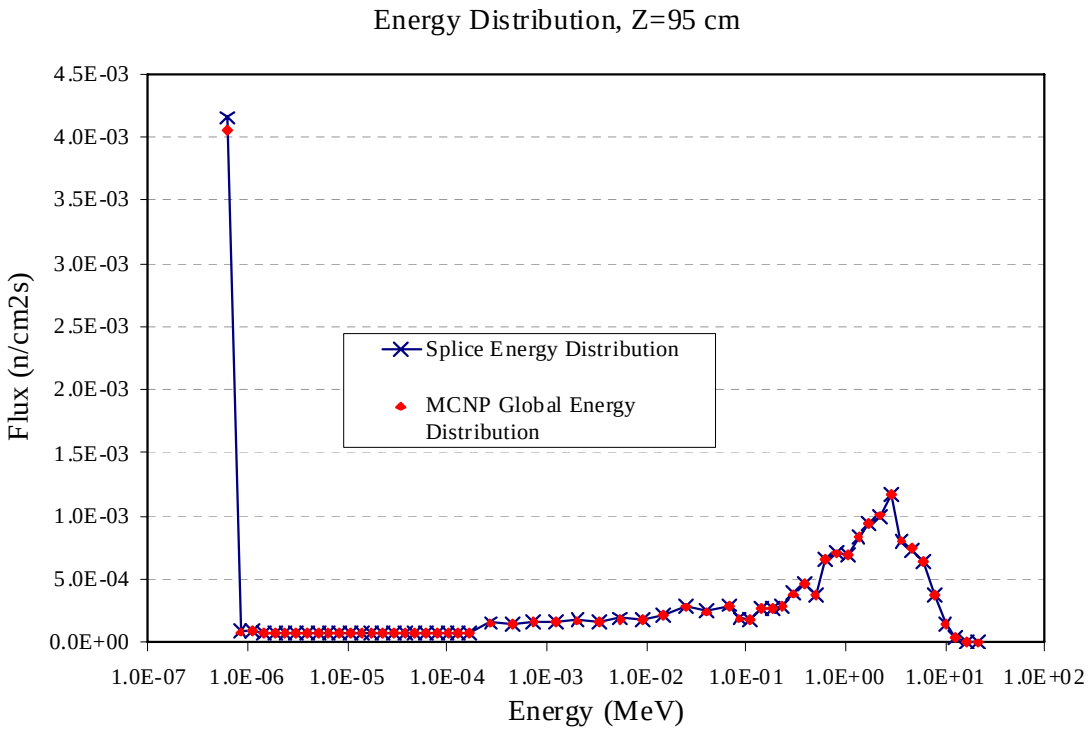

Figure 6. Energy distribution comparison past the steel grid region.

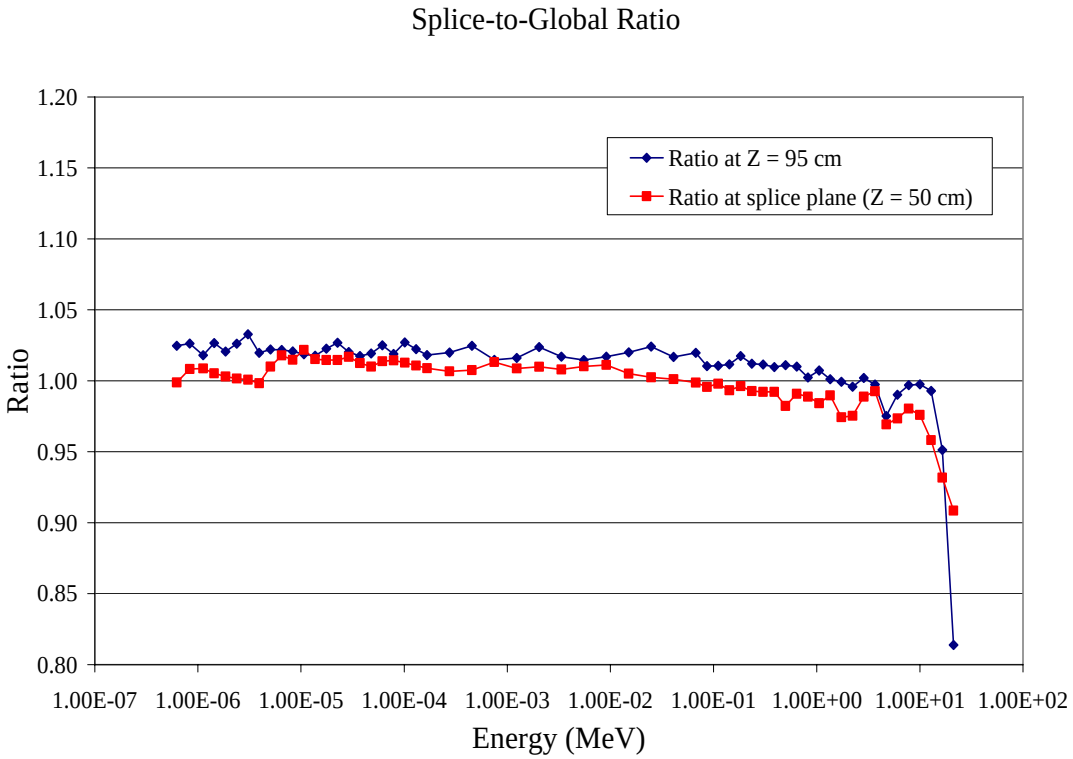

Figure 7. Energy ratio comparison.

Figure 7 shows the ratios between the energy-binned flux distributions illustrated in Figures 5 and 6. One can see the initial source (red line) is sampled very effectively in groups with substantial flux magnitudes. The high energy flux intensity is relatively very low: the error in reproducing the distribution in this energy region results from insufficient sampling and does not significantly alter the total solution. The energy ratio past the steel grid region (blue line) again remains within approximately 2 percent of the global solution everywhere except the very high energy region.

4.2.3. Angular Distribution Comparison

Figure 8 illustrates four cosine-binned MCNP flux tallies taken at the splice boundary. Three are from spliced secondary MCNP solutions and one is from a global MCNP solution. The global MCNP solution shows the most forward-peaked behavior.

Splice Cosine Distribution 1 was generated using a third-order (16 moments) expansion of 144 discrete angles. Splice Cosine Distribution 2 was generated using a fifth-order (36 moments) expansion of 144 discrete angles. Splice Cosine Distribution 3 was generated using a fifth-order (36 moments) expansion of 280 discrete angles.

It is clear from these results that increasing the number of angles in the discrete ordinates quadrature set and the order of the spherical harmonics fit provides closer agreement with a solution fully continuous in angle.

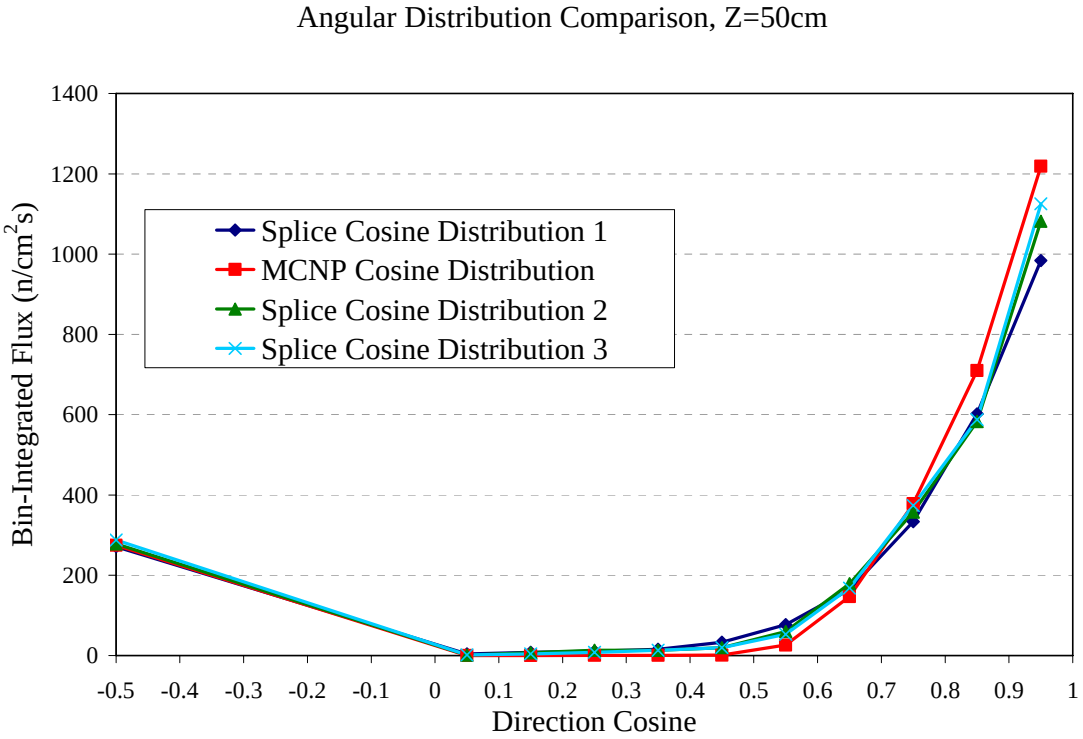

Figure 8. Angular distribution comparison.

5. CONCLUSIONS

A functional PARTISN-to-MCNP splice option using a new method of angular sampling has been successfully developed and tested. We have demonstrated that rejection sampling of an evaluated set of spherical harmonics is a valid method of reproducing discrete angular behavior. We have further shown that increasing the order of the harmonics expansion and/or the number of angles in the quadrature set used in the discrete ordinates problem will produce a more accurate representation of the continuous Monte Carlo solution. Finally, expanding the discrete angular fluxes in spherical harmonics allows us to store only the moments in memory, resulting in a reduction of total storage required to reproduce anisotropic angular behavior.

REFERENCES

1. I. K. Abu-Shumays and C. E. Yehnert, “Angular Interpolations and Splice Options for Three Dimensional Transport Computations,” *Proceedings: Radiation Protection and Shielding 1996 Topical Meeting*, April 21-25, 1996.
2. I.K. Abu-Shumays, “Compatible Product Angular Quadrature for Neutron Transport in x-y Geometry,” *Nucl. Sci. Eng.*, **64**, 299-316 (1977).
3. M.A. Jessee and C.E. Yehnert, “Interpolation Methods and Splice Options For Three-Dimensional Transport Calculations Using PARTISN,” *Proceedings: Mathematics & Computation and Supercomputing in Nuclear Applications, 2007 Topical Meeting*, April 15-19, 2007.
4. R. E. Alcouffe, R. S. Baker, J. A. Dahl, S.A. Turner, and Robert Ward, “PARTISN: A Time-Dependent, Parallel Neutral Particle Transport Code System,” LA-UR-05-3925 (May 2005).
5. X-5 Monte Carlo Team, “MCNP — A General Monte Carlo N-Particle Transport Code, Version 5,” Los Alamos National Laboratory report LA-CP-03-0284 (April 2003).
6. N. N. Lebedev, *Special Functions and Their Applications*, Dover Publications, Inc., Mineola, New York (1972).
7. E. E. Lewis, W. F. Miller, *Computational Methods of Neutron Transport*, American Nuclear Society, Inc,. La Grange Park, Illinois (1993).